- It provides strategic explanation of problem-solving activities.
- It provides generic learning knowledge sources to acquire new control heuristics automatically.
- Its run-time user interface provides capabilities for: displaying knowledge sources, pending actions, and objects on the blackboard; graphically displaying partial solutions via a user-specified interface; recommending pending actions for execution; permitting a user to override a recommendation; executing a designated action; and operating autonomously until a user-specified criterion is met.

BB1 is an evolving system that attempts to incorporate the best results of several research activities. We will continue developing BB1 as a prototype "next-generation" blackboard architecture.

III.C. Administrative Changes

Several administrative changes have occurred over the past year that affect the SUMEX-AIM resource.

In December 1984, the Knowledge Systems Laboratory (KSL) was formed as a reorganization of the Heuristic Programming Project (HPP -- see Appendix A). The new laboratory has a more modular organizational structure that recognizes the broad diversity of work now going on in the KSL and facilitates managing a research group of well over 100 people. The SUMEX-AIM resource continues to play a central role in KSL research.

On January 1, 1985, Mr. Edward Pattermann resigned as SUMEX Director to take a position involving AI tool development at IntelliCorp. He was replaced by Mr. Thomas Rindfleisch, who resumed the Director's role after two years managing the HPP. Mr. Rindfleisch retains his role (20%) as Director of the KSL and Mr. William Yeager has been appointed Assistant SUMEX Director to assist with day-to-day SUMEX management. Mr. Yeager has long been a key technical resource for SUMEX, having developed much of the Ethernet gateway and TIP service now in wide use.

Effective March 1, 1985, Dr. Edward Shortliffe was promoted to Associate Professor of Medicine with tenure. At the same time, Ted was appointed as Principal Investigator of SUMEX and Professor Feigenbaum resumed his role as co-Principal Investigator. This change in PIship in no way affects the long-standing interdisciplinary management of SUMEX, but just gave Dr. Shortliffe the appropriate title to carry out the key scientific and managerial role he already had been playing in SUMEX affairs. His active research has long been a core part of the SUMEX community and his Medical Computer Science group is physically co-located with SUMEX in the Stanford Medical Center. Also, SUMEX is located administratively in the Department of Medicine where Ted has his faculty appointment so he is in an excellent position to effectively represent the project with respect to its relationship with Stanford.

III.D. Resource Management and Allocation

Early in the design of the SUMEX-AIM resource, an effective management plan was worked out with the Biotechnology Resources Program (now Biomedical Research Technology Program) at NIH to assure fair administration of the resource for both Stanford and national users and to provide a framework for recruitment and development of a scientifically meritorious community of application projects. This structure has been described in some detail in earlier reports and is documented in our recent renewal application. It has continued to function effectively as summarized below.

- The AIM Executive Committee meets regularly by teleconference to advise on new project applications, discuss resource management policies, plan workshop activities, and conduct other community business. The Advisory Group meets together at the annual AIM workshop to discuss general resource business and individual members are contacted much more frequently to review project applications. (See Appendix B on page 211 for a current listing of AIM committee membership).

- We have actively recruited new application projects and disseminated information about the resource. The number of formal projects in the SUMEX-AIM community still runs at the capacity of our computing resources. With the development of more decentralized computing resources within the AIM community outside of Stanford (see below), the center of mass of our community has naturally shifted toward the growing number of Stanford applications and core research projects. We still, however, actively support new applications in the national community where these are not able to gain access to suitable computing resources on their own.

- With the advice of the Executive Committee, we have awarded pilot project status to promising new application projects and investigators and where appropriate, offered guidance for the more effective formulation of research plans and for the establishment of research collaborations between biomedical and computer science investigators.

- We have allocated limited "collaborative linkage" funds as an aid to new projects or collaborators with existing projects to support terminals, communications costs, and other justified expenses to establish effective links to the SUMEX-AIM resource. Executive Committee advice is used to guide allocation of these funds.

- We have carefully reviewed on-going projects with our management committees to maintain a high scientific quality and relevance to our biomedical AI goals and to maximize the resources available for newly developing applications projects. Several fully authorized and pilot projects have been encouraged to develop their own computing resources separate from SUMEX or have been phased off of SUMEX as a result and more productive collaborative ties established for others.

- We have continued to provide active support for the AIM workshops. The last one was held at Ohio State University in the summer of 1984 and the next one will be in Washington, DC, hosted by the National Library of Medicine under Drs. Lindberg and Kingsland.

- We have continued our policy of no fee-for-service for projects using the SUMEX resource. This policy has effectively eliminated the serious

administrative barriers that would have blocked our research goals of broader scientific collaborations and interchange on a national scale within the selected AIM community. In turn we have responded to the correspondingly greater responsibilities for careful selection of community projects of the highest scientific merit.

- We have tailored resource policies to aid users whenever possible within our research mandate and available facilities. Our approach to system scheduling, overload control, file space management, etc. all attempt to give users the greatest latitude possible to pursue their research goals consistent with fairly meeting our responsibilities in administering SUMEX as a national resource.

As indicated above, we have sought to retain SUMEX resources for new projects, those exploring new areas in biomedical AI applications and those in such an early state of feasibility that they are unable to afford their own computing resources. This policy has worked effectively as seen from the following lists of terminated projects and projects now using their own computing resources at other sites:

Projects Moved All or In Part to Other Machines:

Stanford Projects:

- GENET [Brutlag, Kedes, Friedland - IntelliCorp]

National Projects:

- Acquisition of Cognitive Procedures (ACT) [Anderson - CMU]
- Chemical Synthesis [Wipke - UC Santa Cruz]
- Simulation of Cognitive Processes [Lesgold - Pittsburgh]
- PUFF [Osborne, Feigenbaum, Fagan - Pacific Medical Center]
- CADUCEUS/INTERNIST [Pople, Myers - Pittsburgh]
- Rutgers [Amarel, Kulikowski, Weiss - Rutgers]
- MDX [Chandrasekaran - Ohio State]
- SOLVER [P. Johnson - University of Minnesota]

Completed Projects Summary

Stanford Projects:

- DENDRAL [Lederberg, Djerassi, Buchanan, Feigenbaum]
- MYCIN [Shortliffe, Buchanan]
- EMYCIN [Shortliffe, Buchanan]
- CRYSALIS [Feigenbaum, Engelmore]
- MOLGEN I [Feigenbaum, Brutlag, Kedes, Friedland]
- AI Handbook [Feigenbaum, Barr, Cohen]

- AGE Development [Feigenbaum, Nii]

National Projects:

- Ventilator Management [Osborne, Feigenbaum, Fagan - Pacific Medical Center]
- Higher Mental Functions [Colby - USC]

III.E. Dissemination of Resource Information

Throughout the history of the SUMEX-AIM resource, we have made extensive efforts at disseminating the AI technology developed here. This has taken the form of many publications -- over 45 combined books and papers are published per year from the KSL; wide distribution of our software including systems software and AI application and tool software, both to other research laboratories and for commercial development; production of films and video tapes depicting aspects of our work; and significant project efforts at studying the dissemination of individual applications systems such as the GENET community (DNA sequence analysis software) and the ONCOCIN resource-related research project (see 102).

Books and Publications

A sampling of the recent research paper publications of the KSL was given in the previous section on core AI research progress. The following lists the major books published in the past 4 years from the KSL:

- *Heuristic Reasoning about Uncertainty: An AI Approach*, Cohen, Pitman, 1985.
- *Readings in Medical Artificial Intelligence: The First Decade*, Clancey and Shortliffe, Addison-Wesley, 1984.
- *Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Project*, Buchanan and Shortliffe, Addison-Wesley, 1984.
- *The Fifth Generation: Artificial Intelligence and Japan's Computer Challenge to the World*, Feigenbaum and McCorduck, Addison-Wesley, 1983.
- *Building Expert Systems*, F. Hayes-Roth, Waterman, and Lenat, eds., Addison-Wesley, 1983.
- *System Aids in Constructing Consultation Programs: EMYCIN*, van Melle, UMI Research Press, 1982.
- *Knowledge-Based Systems in Artificial Intelligence: AM and TEIRESIAS*, Davis and Lenat, McGraw-Hill, 1982.
- *The Handbook of Artificial Intelligence*, Volume I, Barr and Feigenbaum, eds., 1981; Volume II, Barr and Feigenbaum, eds., 1982; Volume III, Cohen and Feigenbaum, eds., 1982; Kaufmann.
- *Applications of Artificial Intelligence for Organic Chemistry: The DENDRAL Project*, Lindsay, Buchanan, Feigenbaum, and Lederberg, McGraw-Hill, 1980.

Software Distribution

We have widely distributed both our system software and our AI tool software. We have no accurate records of the extent of distribution of the system codes because their distribution is not centralized and controlled. The recent programs such as the TOPS-20 file recognition enhancements, the Ethernet gateway and TIP programs, the SEAGATE AppleBus to Ethernet gateway, the PUP Leaf server, the SUMACC development system for Macintosh workstations, and our Lisp workstation programs are well-distributed throughout the ARPANET community and beyond.

We do have reasonably accurate records of the distribution of our AI tool software because the recipient community is more directly coupled to us and the distribution is centralized:

GENET	Prior to the establishment of the BIONET resource at IntelliCorp, we distributed 21 copies of the DNA sequence analysis programs and databases for both DEC-10 and DEC-20 systems.
EMYCIN	A total of 56 sites have received the EMYCIN [4, 34] package for backward-chained, rule-based AI systems.
AGE	The AGE [25] blackboard framework system has been sent out to 35 sites in versions for several machines.
MRS	The MRS [9] logic-based system for meta-level representation and reasoning has been provided to 76 sites.
Other Programs	Smaller numbers of copies of programs such as the SACON [2] knowledge base for EMYCIN, the GLISP [27] system (now distributed by Gordon Novak at the University of Texas), and the new BB1 [14, 13] system have been distributed.

A number of other software packages have been licensed or otherwise made available for commercial development including DENDRAL (Molecular Designs), MAINSAIL (Xidak), UNITS (IntelliCorp), and EMYCIN (Teknowledge and Texas Instruments).

Video Tapes and Films

The KSL and the ONCOCIN project have prepared several video tapes that provide an overview of the research and research methodologies underlying our work and that demonstrate the capabilities of particular systems. These tapes are available through our groups, the Fleischmann Learning Center at the Stanford Medical Center, and the Stanford Computer Forum and copies have been mailed to program offices of our various funding sponsors. The three tapes include:

- *Knowledge Engineering in the Heuristic Programming Project* -- This 20-minute film/tape illustrates key ideas in knowledge-based system design and implementation, using examples from ONCOCIN, PROTEAN, and knowledge-based VLSI design systems. It describes the research environment of the KSL and lays out the methodologies of our work and the long term research goals that guide it.
- *ONCOCIN Overview* -- This is a 30-minute tape providing an overview of the ONCOCIN project. It gives an historical context for the work, discusses the clinical problem and the setting in which the prototype system is being used, and outlines the plans for transferring the system to run on single-user workstations. Brief illustrations of the graphics capabilities of ONCOCIN on a Lisp workstation are also provided.
- *ONCOCIN Demonstration* -- This 1-hour tape provides detailed examples of the key components of the ONCOCIN system. It begins with a demonstration of the prototype system's performance on a time-shared mainframe computer and then shows each of the elements involved in transferring the system to Lisp workstations.

The GENET Dissemination Experiment

Beginning in early 1980, the MOLGEN project investigators at Stanford have made a new set of computing tools available to a national community of molecular biologists through a guest facility called GENET on the SUMEX-AIM resource. This experimental subcommunity was started to broaden MOLGEN's base of scientist collaborators at institutions other than Stanford and to explore the idea of a SUMEX-like resource to disseminate sophisticated software tools to a generally computer-naive community. The enthusiastic response to the very limited announcement of this facility eventually necessitated SUMEX placing severe restrictions on the scope of services provided to this community.

Three main programs were offered to assist molecular genetics users: SEQ, a DNA-RNA sequence analysis program; MAP, a program that assists in the construction of restriction maps from restriction enzyme digest data; and MAPPER, a simplified and somewhat more efficient version of the MOLGEN MAP program, written and maintained by William Pearson of Johns Hopkins University. Some of the other, more-sophisticated programs being developed through MOLGEN research efforts were not yet available for novice users. However, GENET users had access to the SUMEX-AIM programs for electronic messaging, text-editing, file-searching, etc.

The GENET experiment proved so successful that eventually that community was the single biggest consumer of processor cycles on SUMEX. This overload diverted our very limited computing resources away from our mainline goal of supporting projects developing new AI systems in the medical and biological sciences, including molecular biology. Efforts to secure funds to increase SUMEX capacity for the burgeoning GENET use failed. Thus, without any fair way to allocate a small resource to the growing GENET community and in order to restore the necessary emphasis on biomedical computer science research on SUMEX, it was necessary to phase out the GENET usage. We closed the GENET account at the end of 1982, with a mandate from an ad hoc GENET Executive Committee, and phased out all usage by spring of 1983. In the process, we developed procedures by which academic users could obtain their own copies of the GENET programs used at SUMEX and we provided a list of alternate sources for GENET-like computing services. As indicated above, SUMEX has supplied 21 systems to academic users with compatible machines.

Since the phase-out of GENET at SUMEX, IntelliCorp, a commercial AI company, submitted a proposal to the NIH Division of Research Resources for a BIONET resource and was successful in obtaining funding. The BIONET resource began operation in the summer of 1984.

III.F. Suggestions and Comments

Resource Organization

We continue to believe that the Biomedical Research Technology Program is one of the most effective vehicles for developing and disseminating technological tools for biomedical research. The goals and methods of the program are well-designed to encourage building of the necessary multi-disciplinary groups and merging of the appropriate technological and medical disciplines.

Electronic Communications

SUMEX-AIM has pioneered in developing more effective methods for facilitating scientific communication. Whereas face-to-face contacts continue to play a key role, in the longer-term we feel that computer-based communications will become increasingly important to the NIH and the distributed resources of the biomedical community. We would like to see the BRTP take a more active role in promoting these tools within the NIH and its grantee community.

IV. Description of Scientific Subprojects

The following subsections report on the AIM community of projects and "pilot" efforts including local and national users of the SUMEX-AIM facility at Stanford. However, those projects admitted to the National AIM community which use the Rutgers-AIM resource as their home base are not explicitly reported here.

In addition to these detailed progress reports, abstracts for each project and its individual users are submitted on a separate Scientific Subproject Form. However, we have included here briefer summary abstracts of the fully-authorized projects in Appendix C on page 215.

The collaborative project reports and comments are the result of a solicitation for contributions sent to each of the project Principal Investigators requesting the following information:

I. SUMMARY OF RESEARCH PROGRAM
- A. Project rationale
- B. Medical relevance and collaboration
- C. Highlights of research progress
  - --Accomplishments this past year
  - --Research in progress
- D. List of relevant publications
- E. Funding support

II. INTERACTIONS WITH THE SUMEX-AIM RESOURCE
- A. Medical collaborations and program dissemination via SUMEX
- B. Sharing and interactions with other SUMEX-AIM projects
  (via computing facilities, workshops, personal contacts, etc.)
- C. Critique of resource management
  (community facilitation, computer services, communications services, capacity, etc.)

III. RESEARCH PLANS
- A. Project goals and plans
  - --Near-term
  - --Long-range
- B. Justification and requirements for continued SUMEX use
- C. Needs and plans for other computing resources beyond SUMEX-AIM
- D. Recommendations for future community and resource development

We believe that the reports of the individual projects speak for themselves as rationales for participation. In any case, the reports are recorded as submitted and are the responsibility of the indicated project leaders. The only exceptions are the respective lists of relevant publications which have been uniformly formatted for parallel reporting on the Scientific Subproject Form.

IV.A. Stanford Projects

The following group of projects is formally approved for access to the Stanford aliquot of the SUMEX-AIM resource. Their access is based on review by the Stanford Advisory Group and approval by Professor Feigenbaum as Principal Investigator.

In addition to the progress reports presented here, abstracts for each project and its individual users are submitted on a separate Scientific Subproject Form.

IV.A.1. GUIDON/NEOMYCIN Project

GUIDON/NEOMYCIN Project

William J. Clancey, Ph.D.
Department Computer Science
Stanford University

Bruce G. Buchanan, Ph.D.
Computer Science Department
Stanford University

I. SUMMARY OF RESEARCH PROGRAM

A. Project Rationale

The GUIDON/NEOMYCIN Project is a research program devoted to the development of a knowledge-based tutoring system for application to medicine. This work derived from our first system, the MYCIN program. That research led to three sub-projects (EMYCIN, GUIDON, and ONCOCIN) described in previous annual reports. EMYCIN has been completed and its resources reallocated to other projects. GUIDON and ONCOCIN have become projects in their own right.

The key issue for the GUIDON/NEOMYCIN project is to develop a program that can provide advice similar in quality to that given by human experts, modeling how they structure their knowledge as well as their problem-solving procedures. The consultation program using this knowledge is called NEOMYCIN. NEOMYCIN's knowledge base, designed for use in a teaching application, will become the subject material used by a family of instructional programs referred to collectively as GUIDON2. The problem-solving procedures are developed by running test cases through NEOMYCIN and comparing them to expert behavior. Also, we are using NEOMYCIN as a test bed for the explanation capabilities that will eventually be part of our instructional programs.

The purpose of the current contracts is to construct an intelligent tutoring system that teaches diagnostic strategies explicitly. By strategy, we mean plans for establishing a set of possible diagnoses, focusing on and confirming individual diagnoses, gathering data, and processing new data. The tutorial program will have capabilities to recognize these plans, as well as to articulate strategies in explanations about how to do diagnosis. The strategies represented in the program, modeling techniques, and explanation techniques are wholly separate from the knowledge base, so that they can be used with many medical (and non-medical) domains. That is, the target program will be able to be tested with other knowledge bases, using system-building tools that we provide.

B. Medical Relevance and Collaboration

There is a growing realization that medical knowledge, originally codified for the purpose of computer-based consultations, may be utilized in additional ways that are medically relevant. Using the knowledge to teach medical students is perhaps foremost among these, and NEOMYCIN continues to focus on methods for augmenting clinical knowledge in order to facilitate its use in a tutorial setting. A particularly important aspect of this work is the insight that has been gained regarding the need to structure knowledge differently, and in more detail, when it is being used for different purposes (e.g., teaching as opposed to clinical decision making). It was this aspect of the

GUIDON research that led to the development of NEOMYCIN, which is an evolving computational model of medical diagnostic reasoning that we hope will enable us to better understand and teach diagnosis to students. An important additional realization is that these structuring methods are beneficial for improving the problem-solving performance of consultation programs, providing more detailed and abstract explanations to consultation users, and making knowledge bases easier to maintain.

As we move from technological development of explanation and student modeling capabilities, we will in the next year begin to collaborate more closely with the medical community to design an effective, useful tutoring program. Stanford Medical School faculty, such as Dr. Maffly, have shown considerable interest in this project. A research fellow associated with Maffly, Curt Kapsner, M.D., joined the project two years ago to serve as medical expert and liaison with medical students at Stanford.

C. Highlights of Research Progress

C.1 Accomplishments This Past Year

C.1.1 The NEOMYCIN Consultation Program

NEOMYCIN is distinguished from other AI consultation programs by its use of an explicit set of domain-independent metarules for controlling all reasoning. These rules constitute the diagnostic procedure that we want to teach to students: the stages of diagnosis, how to focus on new hypotheses, and how to evaluate hypotheses. This diagnostic procedure as well as the knowledge base underlying the procedure has remained relatively stable this year. Our work in explanation highlighted the importance of making the knowledge used by the system at all levels as explicit as possible. As a result, this year we have extended and refined a previous predicate calculus representation of NEOMYCIN's metaleval rules. To avoid earlier problems of efficiency with this representation, we have also written a compiler that produces Lisp code from our predicate calculus notation. As a result, we are able to run the more efficient Lisp code and use the explicit notation for explanation and modeling.

To develop and test our model of heuristic classification, we are producing from NEOMYCIN a generic system, called HERACLES, that can be used to solve other problems by classification. This is an "E-NEOMYCIN," NEOMYCIN without its current medical knowledge. HERACLES is a variant of EMYCIN; it enables a knowledge engineer to produce NEOMYCIN-like knowledge bases containing the NEOMYCIN diagnostic procedure and domain knowledge organization. To prove its true generality, our first HERACLES knowledge base is in the manufacturing domain, for diagnosing sand casting problems (for the process of forming metal objects using sand molds). Future knowledge bases could be drawn from many medical and non-medical domains.

C.1.2 The ODYSSEUS Modeling System

This effort concerns automation of the transfer of expertise between an expert system and a human expert. A major goal is to produce a system that can watch an expert solve a problem and automatically recognize *differences* between the expert's underlying knowledge base and an expert system's knowledge base. This system should demonstrate how a knowledge of these differences can aid knowledge acquisition and intelligent tutoring. The program implementing this approach, called ODYSSEUS, has several stages of operation. Based on a large set of problem-solving sessions, the program first induces the rule and frame knowledge to drive HERACLES. Using this initial knowledge base as a "half-order theory," subsequent problem-solving sessions are tracked step by step: for each observable step the specialist makes, ODYSSEUS generates and scores the alternative *lines of reasoning* that can explain the specialist's reasoning step. When no plausible reasoning path is found, or all found ones have a low score,

the program assumes it is deficient in either its strategic or domain knowledge. It attempts to acquire the missing knowledge either automatically or by asking the specialist specific questions. In a variation, the specialist justifies each problem-solving step using the vocabulary of an abstract justification language. These justifications aid in scoring alternative plausible lines of reasoning.

Each of the stages of ODYSSEUS has been implemented as a separate subsystem. These subsystems are now being integrated.

C.1.3 The NEOMYCIN Explanation System

The initial explanation system of NEOMYCIN enables the user to ask WHY and HOW questions during a consultation. That is, when the program prompts the user for new data, the user may ask WHY the data is being requested or HOW some strategic task will be (or was) accomplished. Unlike MYCIN's explanation system, upon which this kind of capability is patterned, explanations in NEOMYCIN are in terms of the diagnostic plan, not just specific associations between data and diagnoses.

The next phase of this work is to answer WHY questions by condensing the entire line of reasoning. The program uses general explanation heuristics, models of the user's knowledge of diseases and of strategy, and a history of the user's interaction with the current consultation to select the task, focus, and domain information that is most likely to be of interest. Some of the heuristics used by the explanation system include: 1) mentioning the last task whose focus (or argument) changed in kind (e.g. from a disease hypothesis to a finding request); 2) never mentioning tasks that are merely iterating over a list of rules, findings, or hypotheses; and 3) only mentioning tasks with rules as an argument to programmers. These heuristics, as well as the general procedure for providing explanations, have been implemented in the same task and metarule language used to represent NEOMYCIN's diagnostic strategy. In addition, the explanation system has been extended to use the MRS version of the task metarules. We are thus able to select the specific medical relations that were used by the metarule in determining what action to take. As a result, we have more detailed and concise information to explain to the user. The clearer representation of both the information that can be explained and the explanation procedure provides us with a flexible, explicit encoding of our method for producing explanations, which will serve as a basis for devising tutoring techniques, as well as understanding explanations provided by users of their diagnostic strategy.

Related to our explanation condensation is an effort to teach the strategic language of tasks to students. For example, we will have students annotate a NEOMYCIN transcript in terms of tasks and foci, to help them recognize good strategic behavior. This requires a common language of what the tasks are, e.g. "grouping" and "asking general questions." Rather than just marking annotated tasks, we seek the *principles* by which the tasks could be consistently structured into primitives and auxiliary. These same principles could be used by the explanation system for choosing tasks to mention. Our current theory is that these primitive, or "interesting," operations correspond to metarules that establish a new focus.

C.1.4 Graphics for Teaching

We are continuing to make extensive use of graphics in our programs. As part of our series of instructional programs, GUIDON-WATCH has been implemented as a graphics system for watching NEOMYCIN's reasoning. For example, we can highlight the hypothesis under consideration in the diagnostic taxonomy and show graphically how the program "looks up" its hierarchies before refining hypotheses. In addition, the user is able to explore the findings, hypotheses, rules and tasks that comprise the knowledge base, see selected causal association networks, view the differential as it changes, and keep track of hypotheses with evidence and positive findings. All of these can be easily

selected with a consistent menu system, and windows on the screen are automatically organized to clearly display the information requested by the user.

C.2 Research in Progress

The following projects are active as of June 1984 (see also near-term plans listed in Section III.A):

1. development of a prototype of a bottom-up student modeler
2. standardization of display code
3. prototype of GUIDON-MANAGE
4. prototype of HERACLES and demonstration in non-medical domain
5. user model incorporated in explanations, with summarization
6. student model learning discrepant domain knowledge

D. Publications Since January 1984

1. Clancey, W. J.: *Knowledge acquisition for classification expert systems.* Proc. ACM-84. Also Heuristic Programming Project Report HPP 84-18, Computer Science Dept., Stanford Univ., July, 1984.
2. Clancey, W.J:*Heuristic classification.* Knowledge Systems Laboratory Report KSL 85-5, Computer Science Dept., Stanford University, March 1985.
3. Richer, M., and Clancey, W.J.: *GUIDON-WATCH: A graphic interface for browsing and viewing a knowledge-based system.* Submitted to IEEE.
4. Wilkins, D.C., Buchanan, B.G., and Clancey, W.J.: *Inferring an expert's reasoning by watching.* Proc. 1984 Conference on Intelligent Systems and Machines, Rochester, MI, April 1984, pp.51-58.

II. INTERACTIONS WITH THE SUMEX-AIM RESOURCE

A. Medical Collaborations and Program Dissemination via SUMEX

A great deal of interest in GUIDON and NEOMYCIN has been shown by the medical and computer science communities. We are frequently asked to demonstrate these programs to Stanford visitors or at meetings in this country or abroad. GUIDON is available on the SUMEX 2020. Physicians have generally been enthusiastic about the potential of these programs and what they reveal about current approaches to computer-based medical decision making.

B. Sharing and Interaction with Other SUMEX-AIM Projects

We plan to add learning capabilities of two forms into this framework, involving interactions with the machine learning group within the KSL and Prof. Paul Rosenbloom's project on SOAR.

GUIDON/NEOMYCIN retains strong contact with the ONCOCIN project, as both are siblings of the MYCIN parent. These projects regularly share programming expertise and continue to jointly maintain large utility modules developed for MYCIN. In addition, the central SUMEX development group acts as an important clearing house for solving problems and distributing new methods.

C. Critique of Resource Management

The SUMEX staff has been extremely helpful in maintaining connections between Xerox D-machines and SUMEX. The SUMEX staff also rewrote communication software used to link the D-machines to SAFE, the file saver used by the GUIDON/NEOMYCIN group. This has greatly improved both performance and reliability.

III. RESEARCH PLANS

A. Project Goals and Plans

Research over the next year will continue on several fronts, leading to several prototype instructional programs by early 1986.

1. Test student modeling program on cases chosen for teaching, collecting data for further development of the program, as well as exploring the range of student approaches to diagnosis.

2. Extend the explanation system to do full summaries. Incorporate modeling capabilities that relate inquiries to a user model. Provide explanations tailored to this interpretation of the motivation behind the user's inquiry.

3. Extend student modeling system to include heuristics for generating tests that will confirm and extend the model. Improve the model to include analysis of patterns in model interpretations, including dependency-directed "backtracking" in the belief system and some capability to critique the modeling rules. Relate this to knowledge acquisition research.

4. Work closely with medical students to package NEOMYCIN capabilities in a "workstation" for learning medical diagnosis, determining what mix of student and program initiative is desirable.

5. Refine NEOMYCIN diagnostic model (relations and procedures) by student modeling and knowledge acquisition efforts.

6. Develop, debug, and document an exportable version of HERACLES, a generic knowledge engineering tool that can be used to produce additional medical and non-medical knowledge bases to be tutored by GUIDON2.

7. Formalize heuristics for teaching, given the NEOMYCIN model and heuristics for explanation and modeling, embodied in different versions of GUIDON2.

B. Long term plans: the GUIDON2 Family of Instructional Programs

We sketch here our general conception of the research we plan for 1985-88, specifically the GUIDON2 family of instructional programs, based on the NEOMYCIN problem-solving model. Our ideas are strongly based on recent proposals by J.S. Brown, particularly his paper "Process versus Product -- A perspective on tools for communal and informal electronic learning" and some related papers that he wrote in 1983, in which he proposes methods for giving a student the ability to reflect on how he solved a problem. We have designed a family of seven programs that as a sequence will teach students to think about their own thinking process and to adopt efficient, effective approaches to medical diagnosis.

The key idea is that NEOMYCIN provides a *language* by which a program can converse with a student about strategies and knowledge organization for diagnosis. NEOMYCIN's tasks and structural terms provide the *vocabulary* or *parts of speech*; the meta-rules are the *grammar* of the diagnostic process. We will construct different graphic, reactive environments in which the student can observe, describe, compare, and improve his own diagnostic behavior and that of others. By "reactive environment" we mean that these programs are not passive, they will watch what the student does, build a model of his understanding and learning preferences, and provide corrective advice.

Our approach is to delineate different kinds of interactions that a student might have with a program concerning diagnostic strategies. Thus, each instructional system has a name of the form GUIDON-<student activity>, where the name specifies what the student is doing (e.g., watching, telling). The programs can be made arbitrarily complex by integrating coaches, student models, and explanation systems. There are many shared, underlying capabilities that will be constructed in parallel and improved over time. We try here to separate out these capabilities, trying to get at the minimum interesting activities we might provide for a student.

GUIDON-WATCH The simplest system allows a student to watch NEOMYCIN solve a problem, perhaps one supplied by the student. Graphics display the evolving search space, that is, how tasks, as operators, affect the differential (Differential ---(Question X)---> Differential'). The student can step through slowly and replay the interaction. He can ask for prose explanations and summaries of what the program is doing. The program will also indicate its task and focus for each data request. This introduces the student to the idea that the diagnostic process has structure and follows a certain kind of logic. The graphic capabilities of this program are nearly complete.

GUIDON-MANAGE In this system the student solves a problem by telling NEOMYCIN what task to do at each step. Essentially, the student provides the strategy and the program supplies the tactics (meta-rules) and domain knowledge to carry out the strategy. The program will in general carry through tasks in a logical way, for example, proceeding to test a hypothesis completely, and not "breaking" on low-level tasks that mainly test domain knowledge rather than strategy. The program will not pursue new hypotheses automatically. However, the student will always see what questions a task caused the program to request, as well as how the differential changes. This activity leads the student to observe what a strategy entails, helping him become a better observer of his own behavior. Here he shows that he knows the structural vocabulary that makes a strategy appropriate.

GUIDON-ANNOTATE This system allows the student to annotate a NEOMYCIN typescript, *explaining* in strategic and/or domain terms what the program is doing each time it requests new case data, indicating the task and focus associated with each data request. The program will indicate, upon request, where the student is incorrect and which annotations are different from NEOMYCIN's, but are still reasonable interpretations. The student will be able to choose these tasks from a menu of icons, either linearly or hierarchically displayed, as he prefers. (Again, NEOMYCIN will

annotate its own solutions upon request and allow replaying.) This activity gets the student to think strategically by recognizing a good strategy. In this way, he learns to recognize how strategies affect the problem space.

GUIDON-APPRENTICE This is a variant of NEOMYCIN in which the program stops during a consultation and asks the student to propose the next data request(s). The student is asked to indicate the task and focus he has in mind, plus the differential he is operating upon. The program compares this proposal to what NEOMYCIN would do. In this activity we descend to the domain level and require the student to instantiate a strategy appropriately. Ultimately, such a program will use a *learning model* that anticipates what the student is ready to learn next and how he should be challenged. Early versions can simply use built-in breakpoints supplied by an expert teacher. In the future, programs will develop their own curriculums from a case library.

GUIDON-DEBUG Here the student is presented with a buggy version of NEOMYCIN and must debug it. He goes through the steps of annotating the buggy consultation session, indicating what questions are out of order or unnecessary, indicating what tasks are not being invoked properly, and then trying out his hypothesis on a "repaired" system. He is asked to predict what will be different, then allowed to observe what happens. This activity teaches the student to recognize how a diagnostic solution can be non-optimal, further emphasizing the value of good strategy. It also provides him with key meta-cognitive practice for criticizing and debugging problem behavior. With time, GUIDON will collect examples of buggy student behavior, providing a library of pitfalls to be shown to new students.

GUIDON-SOLVE This is the complete tutorial system. The student carries through diagnosis completely, while a student modeling program attempts to track what he is doing and a coach interrupts to offer advice. Here annotation, comparison, debugging, and explanation are all integrated to illustrate to the student how his solution is non-optimal. For example, the student might be asked to annotate his solution after he is done; this will point out strategic gaps in his awareness and provide a basis for critique and improvement. A "curriculum" based on frequent student faults and important things to learn will drive the interaction. In this activity, the student is on his own. Faced with the proverbial "blank screen," he must exercise his diagnostic procedure from start to finish.

GUIDON-GAME Two or more students play this together on a single machine. They are given a case to solve together, and each student requests data in turn. All students receive the requested information. When a student is ready, he makes a diagnosis, indicated secretly to the program while the others are not watching. He then drops out of the questioning sequence. However, he can re-enter later, but of course will be penalized. Afterwards, score is based on the number of questions asked and use of good strategy. The coach will indicate to weak players what they could learn from strong players, encouraging them to discuss certain issues among themselves. *Variation*: one person solves while one or more competing students annotate the solution and show where it could be improved. *Variation*: one team introduces a bug into NEOMYCIN (and predicts the effect), and the other team finds it (as in SOPHIE). This activity will encourage students to share their experiences and talk to and learn from each other.

C. Requirements for Continued SUMEX Use

Although most of the GUIDON and NEOMYCIN work is shifting to Xerox Dolphins and Dandelions (D-machines), the DEC 2060 and 2020 continue to be key elements in our research plan. Our primary use of the 2060 will be to develop the NEOMYCIN consultation system, possibly by remote ARPANET access. Because of address space limitations, the consultation program can be combined with explanation or student modeling facilities, but not both, as is required for GUIDON2 programs. We continue to use the 2020 for demonstrating the original GUIDON program. As always, the 2060 will be essential for work at home, writing, and electronic mail.

D. Requirements for Additional Computing Resources

With the addition of two new D-machines for this work, our computing needs will be adequately met in the coming 1-2 years at least.

The D-machine's large address space permits development of the large programs that complex computer-aided instruction requires. Graphics enable us to develop new methods for presenting material to naive users. We also plan to use the D-machine as a reliable, constant "load-average" machine, for running experiments with physicians and students. The development of GUIDON2 on the D-machine will demonstrate the feasibility of running intelligent consultation or tutoring systems on small, affordable machines in physicians' offices, schools, and other remote sites.

E. Recommendations for Future Community and Resource Development

As we shift our development of systems to personal Lisp machines, such as the Dolphin, it becomes more difficult to access these programs remotely for access from our homes (so that we may work conveniently during the evenings and weekends) and from remote sites for collaboration and demonstration. This problem will be partly ameliorated by "dial-up" (modem) access to these machines, but the use of bitmapped displays requiring a high bandwidth makes the phone lines inadequate for our purposes. Further technological development of networks, probably involving access over cables, will be necessary.

As computer resources become more distributed, the need for a central machine does not diminish. Programs and knowledge bases continue to be shared, requiring high-speed network connections among computers and file servers. SUMEX-AIM's role will shift slightly over the next few years to accommodate these needs, but its identity as a central resource will only change in kind, not importance. Moreover, sophisticated printing devices, such as the Xerox RAVEN, must necessarily be shared, again using a network. Maintenance of this network and its shared devices will become a key activity for the SUMEX staff. Thus, while computing resources will be provided by the "outboard engines" of personal machines, the community will remain intricately linked and dependent on common, but peripheral, resources.

From this perspective, future resource development should focus on improving the capabilities of networks, file servers, and attached devices to respond to individual requests. Multi-processing becomes a necessity in such an environment, so a request can be honored while the user returns to continue his programming or editing.

IV.A.2. MOLGEN Project

MOLGEN - Applications of Artificial Intelligence to Molecular Biology: Research in Theory Formation, Testing, and Modification

Prof. E. Feigenbaum and Dr. P. Friedland
Department of Computer Science
Stanford University

Prof. Charles Yanofsky
Department of Biology
Stanford University

I. SUMMARY OF RESEARCH PROGRAM

A. Project Rationale

The MOLGEN project has focused on research into the applications of symbolic computation and inference to the field of molecular biology. This has taken the specific form of systems which provide assistance to the experimental scientist in various tasks, the most important of which have been the design of complex experiment plans and the analysis of nucleic acid sequences. Our current research concentrates on scientific discovery within the subdomain of regulatory genetics. We desire to explore the methodologies scientists use to modify, extend, and test theories of genetic regulation, and then emulate that process within a computational system.

Theory or model formation is a fundamental part of scientific research. Scientists both use and form such models dynamically. They are used to predict results (and therefore to suggest experiments to test the model) and also to explain experimental results. Models are extended and revised both as a result of logical conclusions from existing premises and as a result of new experimental evidence.

Theory formation is a difficult cognitive task, and one in which there is substantial scope for intelligent computational assistance. Our research is toward building a system which can form theories to explain experimental evidence, can interact with a scientist to help to suggest experiments to discriminate among competing hypotheses, and can then revise and extend the growing model based upon the results of the experiments.

The MOLGEN project has continuing computer science goals of exploring issues of knowledge representation, problem-solving, discovery, and planning within a real and complex domain. The project operates in a framework of collaboration between the Heuristic Programming Project (HPP) in the Computer Science Department and various domain experts in the departments of Biochemistry, Medicine, and Biology. It draws from the experience of several other projects in the HPP which deal with applications of artificial intelligence to medicine, organic chemistry, and engineering.

B. Medical Relevance and Collaboration

The field of molecular biology is nearing the point where the results of current research will have immediate and important application to the pharmaceutical and chemical industries. Already, clinical testing has begun with synthetic interferon and human growth hormone produced by recombinant DNA technology. Governmental reports estimate that there are more than 200 new and established industrial firms already undertaking product development using these new genetic tools.

The programs being developed in the MOLGEN project have already proven useful and important to a considerable number of molecular biologists. Currently several dozen researchers in various laboratories at Stanford (Prof. Paul Berg's, Prof. Stanley Cohen's, Prof. Laurence Kedes', Prof. Douglas Brutlag's, Prof. Henry Kaplan's, and Prof. Douglas Wallace's) and over 400 others throughout the country have used MOLGEN programs over the SUMEX-AIM facility. We have exported some of our programs to users outside the range of our computer network (University of Geneva [Switzerland], Imperial Cancer Research Fund [England], and European Molecular Biology Institute [Heidelberg] are examples). The pioneering work on SUMEX has led to the establishment of a separate NIH-supported facility, BIONET, to serve the academic molecular biology research community with MOLGEN-like software. BIONET is now serving many of the computational needs of over 1000 academic molecular biologists in the United States.

C. Highlights of Research Progress

C.1 Accomplishments

The current year has seen the completion of our initial study of the Yanofsky project on genetic regulation in the *trp* operon. In addition we have tested several models of qualitative simulation of biological systems and begun our design of a theory discovery system. Finally, a new application program for DNA sequence analysis was developed by one of our research collaborators. The highlights of this work are summarized in several categories below.

C.1.1 The Scientific Process of Theory Formation, Modification, and Testing

The first goal of our work in scientific theory discovery was to extensively study an existing example of the process. Professor Charles Yanofsky's work in elucidating the structure and function of regulation in the trp operon of E. coli provided us with an excellent subject that spanned twelve years of research, dozens of collaborators, and almost one hundred research papers.

We have conducted extensive interviews with Professor Yanofsky and many of his former students and collaborators. We have examined most of the relevant research papers. We believe we now have a good understanding of the three major classes of knowledge that were important in the discovery of the theory of regulation in the trp operon: knowledge about the relevant biological objects, knowledge about the techniques used to elicit new information, and discovery heuristics used to build new models.

In addition, we have developed an initial model for the inference mechanisms used during the discovery process. This model includes at least four different types of reasoning: data-driven, theory-driven, analogy to closely-related biological systems, and analogy to other systems (railroad engines and tracks, for example).

C.1.2 Knowledge-Based Simulation of the Trp Operon

The first major programming task of our project was to build a knowledge base representing the initial state of knowledge about the tryptophan operon system at the beginning of the Yanofsky research. This initial knowledge base contains information relevant to genetic regulation in general and to the trp operon system in particular. The information relates both to structure, i.e. the physical characteristics of the biological objects, and to function, i.e. the operational characteristics of the biological objects. In addition, the procedural knowledge needed to relate structure to function plays an important part in the knowledge base.

The goal was to have a knowledge base that can be used "actively" to simulate the result of various possible changes in the underlying regulatory model. For example, a

common experimental method for studying a biological system is to introduce a mutation which destroys the functionality of some piece of the system. The regulatory knowledge base should be able to simulate and describe the results of such a "deletion mutation."

As a first experiment, we built the knowledge base using the Unit System (developed under previous MOLGEN work). We were able to successfully model most of the important processes of Jacob-Monod repression, the initial model of genetic regulation used in the Yanofsky research.

C.1.3 A Model for Theory Discovery

In parallel with our work on knowledge base construction, we designed an initial architecture for theory proposal, extension, and correction. In human scientists we have observed at least four major types of reasoning during the cognitive process. The first is data-driven reasoning when the major goal is to explain individual experimental results. The second is theory-driven reasoning which occurs when a partial theory or model drives its own extension. The third type of reasoning involves looking at closely related biological systems (e.g, noticing a similar behavior in the his operon system). The final type of reasoning relates to more distant analogies; thinking of DNA polymerase moving along a nucleotide sequence as similar to a railroad engine moving along a set of tracks. Our discovery system architecture embraces all of these reasoning types within a blackboard-style hybrid architecture.

In addition, we have fit our overall model of simulation and discovery into a framework of research on machine learning. This framework involves interacting performance and learning elements. The performance element, here the knowledge-based system for qualitative simulation of regulatory genetics, is asked to explain observations from the real world. The learning element, here the discovery architecture described above, is able to evaluate the explanations and "tune" the performance element by changing its model (or theory) of the world.

C.1.1.4 Simultaneous alignment of DNA sequences--MULTAN

Previously, MOLGEN researchers have developed numerous programs to aid in the symbolic analysis of DNA sequences. During the last year Dr. William Bains (a postdoctoral scholar in Professor Kedes' laboratory), completed a program called MULTAN which allows the facile alignment of three or more DNA sequences. This was a major unsolved problem in sequence analysis and the program is now undergoing final testing on the BIONET resource. In the future, we expect that BIONET will support development of application-oriented programs of this type, while MOLGEN and SUMEX will focus on research-oriented systems with major AI goals.

C.2 Research in Progress

We have two major goals over the next several months. The first is to convert and enhance our knowledge-based simulation model within the KEE tool from IntelliCorp, Inc. KEE will be a significant improvement over the Unit System in three areas: speed, functionality, and support. IntelliCorp is providing KEE for use in our research without charge. Studies have indicated that using KEE will unable us to produce a reasonable prototype of our discovery system in about half the time or using the Unit System. Our second goal is to more formally define the learning element of our discovery system and to build a first test system that operates upon the simulation system knowledge base.

D. Publications

1. Bach, R., Friedland, P., Brutlag, D. and Kedes, L.: *MAXIMIZE, a DNA sequencing strategy advisor.* Nucleic Acids Res. 10(1):295-304, January, 1982.

2. Bach, R., Friedland, P., and Iwasaki, Y.: *Intelligent computational assistance for experiment design.* Nucleic Acids Res. 12(1):11-29, January, 1984.

3. Brutlag, D., Clayton, J., Friedland, P. and Kedes, L.: *SEQ: A nucleotide sequence analysis and recombination system.* Nucleic Acids Res. 10(1):279-294, January, 1982.

4. Clayton, J. and Kedes, L.: *GEL, a DNA sequencing project management system.* Nucleic Acids Res. 10(1):305-321, January, 1982.

5. Feitelson, J. and Stefik, M.J.: *A case study of the reasoning in a genetics experiment.* Heuristic Programming Project Report HPP-77-18 (working paper), May, 1977.

6. Friedland, P.: *Knowledge-based experiment design in molecular genetics.* Proc. Sixth IJCAI, August, 1979, pp. 285-287.

7. Friedland P.: *Knowledge-based experiment design in molecular genetics.* Stanford Computer Science Report STAN-CS-79-760 (Ph.D. thesis), December, 1979.

8. Friedland, P., Kedes, L. and Brutlag D.: *MOLGEN--Applications of symbolic computation and artificial intelligence to molecular biology.* Proc. Battelle Conference on Genetic Engineering, April, 1981.

9. Friedland, P.: *Acquisition of procedural knowledge from domain experts.* Proc. Seventh IJCAI, August, 1981, pp. 856-861.

10. Friedland, P., Kedes, L., Brutlag, D., Iwasaki, Y. and Bach R.: *GENESIS, a knowledge-based genetic engineering simulation system for representation of genetic data and experiment planning.* Nucleic Acids Res. 10(1):323-340, January, 1982.

11. Friedland, P., and Kedes, L.: *Discovering the secrets of DNA.* (To appear in a joint issue of Communications of the ACM and IEEE/Computer, ~~October,~~ 1985). CACM 28(11): 1164-1186

12. Friedland, P. and Iwasaki Y.: *The concept and implementation of skeletal plans.* (To appear in Journal of Automated Reasoning, Vol. 1, No. 2, 1985).

13. Friedland, P., Armstrong, P., and Kehler, T.: *The role of computers in biotechnology.* BIO\TECHNOLOGY 565-575, September, 1983.

14. Iwasaki, Y. and Friedland, P.: *SPEX: A second-generation experiment design system.* Proc. of Second National Conference on Artificial Intelligence, August, 1982, pp. 341-344.

15. Martin, N., Friedland, P., King, J. and Stefik M.J.: *Knowledge base management for experiment planning in molecular genetics.* Proc. Fifth IJCAI, August, 1977, pp. 882-887.

16. Meyers, S. and Friedland, P.: *Knowledge-based simulation of regulatory genetics in bacteriophage Lambda.* Nucleic Acids Res. 12(1):1-9, January, 1984.

17. Stefik, M. and Friedland, P.: *Machine inference for molecular genetics: Methods and applications.* Proc. of NCC, June, 1978.

18. Stefik, M.J. and Martin N.: *A review of knowledge based problem solving as a basis for a genetics experiment designing system.* Stanford Computer Science Report STAN-CS-77-596, March, 1977.

19. Stefik, M.: *Inferring DNA structures from segmentation data: A case study.* Artificial Intelligence 11:85-114, December, 1977.

20. Stefik, M.: *An examination of a frame-structured representation system.* Proc. Sixth IJCAI, August, 1979, pp. 844-852.

21. Stefik, M.: *Planning with constraints.* Stanford Computer Science Report STAN-CS-80-784 (Ph.D. thesis), March, 1980.

II. INTERACTIONS WITH THE SUMEX-AIM RESOURCE

SUMEX-AIM continues to provide the bulk of our computing resources. The facility has not only provided excellent support for our programming efforts but has served as a major communication link among members of the project. Systems available on SUMEX-AIM such as INTERLISP, TV-EDIT, and BULLETIN BOARD have made possible the project's programming, documentation and communication efforts. The interactive environment of the facility is especially important in this type of project development.

We strongly approve of the network-oriented approach to a programming environment that SUMEX has begun to evolve into. The ability to utilize LISP workstations for intensive computing while still communicate with all of the other SUMEX resources has been very valuable to our work. We see a satisfactory mode of operation where most programming takes place on the workstations and most electronic communications, information sharing, and document preparation takes place within the mature TOPS-20 environment. The evolution of SUMEX has alleviated most of our previous problems with resource loading and file space. Our current workstations are not quite fast nor sophisticated enough, but we are encouraged by the progress that has been made.

We have taken advantage of the collective expertise on medically-oriented knowledge-based systems of the other SUMEX-AIM projects. In addition to especially close ties with other projects at Stanford, we have greatly benefited by interaction with other projects at yearly meetings and through exchange of working papers and ideas over the system.

The ability for instant communication with a large number of experts in this field has been a determining factor in the success of the MOLGEN project. It has made possible the near instantaneous dissemination of MOLGEN systems to a host of experimental users in laboratories across the country. The wide-ranging input from these users has greatly improved the general utility of our project.

We find it very difficult to find fault with any aspect of the SUMEX resource

management. It has made it easy for us to expand our user group, to give demonstrations (through the 20/20 adjunct system as well as the LISP workstations), and to disseminate software to non-SUMEX users overseas.

III. RESEARCH PLANS

A. Project Goals And Plans

Our current work has the following major goals:

1. Use the knowledge base to explain observations that are indeed explainable without changes to the current model. For example, "I have observed a mutation that causes constitutive (uncontrolled) production of tryptophan. How can that be explained within the Jacob-Monod model?" This process will be accomplished by some combination of forward simulation and backward rule-chaining.

2. Begin to recognize when observations are "interesting." Interesting here has one of the following broad meanings:

   a. A seeming direct contradiction to the existing theory.

   b. A statistically rare occurrence (one that is understandable by the current theory, but should not occur very often).

   c. A dramatic confirmation of the existing model.

   d. An observation currently unpredictable by the current model because the model is either not detailed enough or incomplete. The observation in this case must have a relation to the model because an important object of the model is involved or it relates to an effect predicted by the model.

3. Build a mechanism for postulating extensions or corrections to the current theory: a contrained regulatory theory generator. The overall approach to this mechanism is perhaps the most interesting problem in our work. In discussions with other computer scientists, the notion of "or" reasoning where the theory construction process consists of hierarchical refinement of abstract ideas into more detailed ones, and "and" reasoning where the theory is built up in little pieces at many different levels simultaneously has emerged. We see strong evidence for both types of reasoning within Yanofsky's project. In fact, as stated above, the global model of Yanofsky's laboratory is a hybrid one. Individual graduate students performed "and" tasks--filling in details of seemingly unrelated pieces of the model. Yanofsky was the master "or" reasoner, slowly building a hierarchical model of the new regulatory mechanism. It is in this area of our research where the greatest discussion with AI colleagues is needed and which may produce the most significant AI benefits.

4. Build a mechanism for evaluating alternative theories. This would include rating the theories based on plausibility, selectability, completeness, significance, and so on. We hope the evaluation process produces information useful in discriminating among the possible theories.

5. Test the entire structure on the evolving trp operon regulatory system. Experiment with different initial knowledge bases to see how the discovery process is altered by the availability of new techniques, analogous systems, etc.